AFRL-VA-WP-TR-2007-3054

INTELLIGENT FLIGHT CONTROL SIMULATION RESEARCH PROGRAM

Brian Stolarik

Institute for Scientific Research, Inc.
1000 Galliher Drive
Fairmont, WV 26554

FEBRUARY 2007

Final Report for 01 December 2002 – 30 September 2006

STINFO COPY

AIR VEHICLES DIRECTORATE
AIR FORCE MATERIEL COMMAND
AIR FORCE RESEARCH LABORATORY
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7542

NOTICE AND SIGNATURE PAGE

AFRL-VA-WP-TR-2007-3054 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

*//Signature//   //Signature//

HOWARD T. EMSLEY   GARY K. HELLMANN, Chief
Technical Area Leader   Aerospace Vehicles Technology
                                                                     Assessment & Simulation Branch

//Signature//

JEFFREY C. TROMP
Senior Technical Advisor
Control Sciences Division
Air Vehicles Directorate

1. REPORT DATE *(DD-MM-YY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
February 2007	Final	12/01/2002 – 09/30/2006

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
INTELLIGENT FLIGHT CONTROL SIMULATION RESEARCH PROGRAM	FA8650-04-C-3402
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER 62201F

6. AUTHOR(S)	5d. PROJECT NUMBER
Brian Stolarik	A052
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER 0B

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Institute for Scientific Research, Inc. 1000 Galliher Drive Fairmont, WV 26554	

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY ACRONYM(S)
Air Vehicles Directorate Air Force Research Laboratory Air Force Materiel Command Wright-Patterson Air Force Base, OH 45433-7542	AFRL-VA-WP
	11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S) AFRL-VA-WP-TR-2007-3054

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

13. SUPPLEMENTARY NOTES
Report contains color. PAO Case Number: AFRL/WS 07-1403, 13 Jun 2007.

14. ABSTRACT

Between December 2002 and September 2006, the Institute for Scientific Research, Inc. (ISR) performed basic and applied research for the Air Force Research Laboratory/Control Sciences Division's (AFRL/VAC). Under the program, entitled "Intelligent Flight Control Simulation Research Laboratory," a variety of technologies were investigated or developed during the course of the research for AFRL/VAC.

The majority of scientific and technical deliverables as a result of the research activities were open literature publications, periodic research reports, customer briefing materials, customer briefings, simulation software and simulation software manuals. This report summarizes the Automated Aerial Refueling, and Multiple Agent Coordinated Control efforts.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT:	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON (Monitor)
a. REPORT Unclassified	b. ABSTRACT Unclassified	c. THIS PAGE Unclassified	SAR	22	Howard T. Emsley 19b. TELEPHONE NUMBER *(Include Area Code)* N/A

TABLE OF CONTENTS

1 INTRODUCTION

Between December 2002 and September 2006, the Institute for Scientific Research, Inc. (ISR) performed basic and applied research for the Air Force Research Laboratory/Control Sciences Division's (AFRL/VAC). The research was funded by annual congressional additions to the AFRL budget. Under the program, entitled "Intelligent Flight Control Simulation Research Laboratory,". a variety of technologies at various Technology Readiness Levels (TRL) were investigated or developed during the course of the research for three branches within AFRL/VAC: VACA, VACC, and VACD. This report serves as the contractually mandated Final Report for those efforts.

The majority of scientific and technical deliverables as a result of the research activities were open literature publications, periodic research reports, customer briefing materials, customer briefings, simulation software and simulation software manuals. This report will only summarize each of the research topics investigated since the technology program managers and principal investigators in each of the three branches are in possession of the deliverables for their technology area.

2 RESEARCH SUMMARY

2.1 Automated Aerial Refueling

The Automated Aerial Refueling (AAR) program within VACC is chartered to develop a system to enable an unmanned aircraft to autonomously refuel during flight from a KC-135 tanker without requiring changes to either the tanker or standard refueling operations regulations. ISR was tasked to investigate the feasibility of using a passive sensor system on the UAV to perform the station keeping maneuver. The efforts for this task are summarized below.

2.1.1 Simulation Development

In order to study the AAR problem, a detailed modeling and simulation environment was needed. ISR subcontracted the majority of that development to West Virginia University (WVU). WVU built an AAR modeling and simulation environment in MATLAB/Simulink that incorporated a simulated tanker[a], simulated Innovative Control Effectors (ICE) receiver vehicle[b], and a simulated visible spectrum camera system. Additionally, a specific control scheme based on sensor fusion between Global Positioning System (GPS) based and a Machine Vision (MV) based measurement was incorporated. This simulation environment was used for the duration of the contract to test different AAR concepts and conduct preliminary feasibility analysis of the passive sensor based approach to AAR.

---

[a] A model of the Boeing 747 model was used in lieu of a KC-135 model, which was not available

[b] The target receiver vehicle for the AAR program was the Joint Unmanned Combat Aerial System (JUCAS), but a model was not available at the time.

1

2.1.2 Machine Vision Research

The majority of ISR's efforts on the AAR program dealt with the evaluation of various MV algorithms to test their ability to identify, extract, and track features on the KC-135 tanker from a video stream. These algorithms comprise the feature identification function, which is responsible for finding the pixel coordinates for each feature in each video frame and passing those feature coordinates to the pose estimation function. Trade studies were designed and completed for feature identification of visible light spectrum video at contact position. The results of those trade studies were presented at the March 2006 AAR Technical Interchange Meeting (TIM).

Early efforts to understand the MV challenges in the AAR application were limited by the lack of actual flight video of the operation. ISR used the WVU-built simulation environment to create simulated video data to test the various methods of feature extraction. Additionally, to have the simulation fully capable, effort was also focused on finding a suitable pose estimation routine. Pose estimation is a function in the Hybrid System that translates detected tanker features from two dimensions (camera frame) to three dimensions. Finding the appropriate pose estimation function was the responsibility of Northrop Grumman Corporate (NGC), who ISR did not have a working relationship with at the time. Once NGC and ISR began working together on the Hybrid Solution, NGC quickly realized the performance benefits of the pose estimation chosen for the ISR simulation and adopted it into their trade studies. The routine, Lu, Hager and Mjolsness (LHM), turned out to be the victor in the NGC trade studies.

In August 2004, AAR flight tests were conducted and flight video captured from a single EO black and white camera mounted on the front of the Learjet. Trade studies were performed on this flight video to assess the feasibility of using visible light spectrum imagery to extract, identify and track KC-135 surface features from the receive aircraft contact position. The focus of the trade studies was the evaluation of general feature finding algorithms; no algorithm design work was undertaken. Rather, the trade studies sought to understand if well-known and well-understood general feature finding algorithms would be applicable. The trade studies were built around an image processing concept that a feature extraction algorithm could find small surface features (e.g., corners or edges) and a feature association algorithm could then associate those features with a known feature set model in order to know which of the features found were features of interest on the tanker and which features were to be disregarded.

The research efforts up to October 2005 limited the feature extraction algorithms to two candidates, Harris and SUSAN. For the purposes of comparison, both SUSAN and Harris were evaluated in two major categories: Functional, which assessed the overall ability of the algorithm to detect features, and Physical, which looked at the computational requirements of the algorithms. Since the primary focus of the trade studies were feature extraction routines, feature association was not fully investigated and only a Euclidean distance based approach was evaluated.

The outcome of the feature extraction trade study showed that the Harris algorithm performed better than the SUSAN algorithm; however, Harris requires marginally more

computational overhead. Both algorithms are extremely sensitive to a threshold that would have to be automatically tuned in flight in an operational system, which is less than desirable. The feature association routine evaluated does have potential; however, other association algorithms need to be investigated.

The results of the visible light spectrum trade studies, as well as concerns about using visible light spectrum cameras during poor weather or night operations, supported the move to evaluating the infrared spectrum to achieve passive sensing AAR. Data collection flight tests were conducted in August 2006, but that data had not been made available to ISR for evaluation at the conclusion of the subject research contract.

2.1.3 Learjet Heads Down Display

ISR was tasked with developing a simulated Heads Down Display (HDD) with full instrumentation to emulate the Learjet cockpit being used in the AAR program as the surrogate JUCAS. The HDD was to be used in AFRL/VACD's Infinity Cube Flight Simulator to aid in AAR test simulations.

Development began by gathering CALSPAN Learjet pilot requirements during a visit to CALSPAN in summer 2005. The pilot only requested information from six sensors during the AAR operation: airspeed (CAS), compass (HSI), altitude direction instrument (ADI), altimeter, vertical speed indicator (IVSI), and the engage panel. Also during the visit to CALSPAN, digital photos of the Learjet instrument panel were taken to be used for the HDD simulation.

Initial development used the FDASH framework software on LINUX based machines. The approach used consisted of removing the functional components from the digital images that were taken and replacing them by using FDASH. OpenGL and C++ programming languages were also used in the manipulation and functionality of the instruments. The development phase for the FDASH release took from June – October 2005 when this initial HDD software release was delivered and installed in AFRL/VACD's Infinity Cube.

Because FDASH was phased out of AFRL/VACD in fall 2005 and replaced with EAAGLES, the HDD simulation software needed to be ported to EAAGLES. Additionally, new requirements for the HDD simulation were levied to enable the pilot to engage/disengage the AAR flight control laws, to set the course and heading for the compass, and to turn off the safety trip button once a safety trip occurs. Development and testing activities for these new requirements, as well as porting from FDASH to EAAGLES, occurred November 2005 – March 2006. Installation of this second and final release of the HDD simulation software occurred on 14 March 06.

2.2 Multiple Agent Coordinated Control

2.2.1 Simulation Development

MultiUAV is a multiple UAV mission management simulation software package developed by AFRL in the MATLAB/Simulink environment to aid in the investigation of multiple agent coordinated control strategies. ISR assisted AFRL in expanding MultiUAV's capabilities by adding simulation interoperability software, introducing new mission scenarios, and researching cooperative control teaming strategies.

Prior to modification by ISR, MultiUAV simulated a Wide Area Search and Destroy (WASD) mission. In this mission a group of Low Cost Autonomous Attack System (LOCAAS) munitions search an area with the goal to find, classify, attack and perform battle damage assessment on any hostile targets. AFRL tasked ISR with leveraging this simulation capability with other various simulation capabilities within AFRL/VA to increase the simulation fidelity. ISR's incorporation of the DoD standard High Level Architecture (HLA) interface into MultiUAV enabled MultiUAV to easily interoperate with other AFRL simulation packages. It also allowed MultiUAV simulation components to be distributed to multiple computers and enabled investigation of UAV communications from entities outside the MATLAB/Simulink domain. HLA was chosen due to its selection by the DoD as the architecture protocol for distributed simulations and IEEE 1516 standardization.

HLA Interface for MultiUAV

To incorporate the HLA capability, MultiUAV's inter-vehicle communication model was disabled and replaced with a series of Simulink S-Functions. This first produced a specialized set of functions that were only reconfigured through compilation and linking of C++ code. The overall goal of the initial software effort was to seamlessly allow UAV team members to communicate through HLA messages without affecting other components of the vehicle, such as path planning, sensor modeling, and assignment scheduling.

The initial MultiUAV-HLA integration was demonstrated to AFRL/VA in June 2003. During this briefing [1], ISR received approval of the HLA design and additional requirements based on preliminary results of the software interface. AFRL/VA requested that the HLA interface capability be integrated into the first major revision of the tool (MultiUAV 2.0). An external HLA interface module was also requested to help study UAV cooperative control affects in the presence of inter-vehicular communication noise, latency, and dropouts.

In October 2003, ISR demonstrated MultiUAV simulating a WASD mission using HLA for the inter-vehicle communication. During this briefing [2], two major objectives were achieved:

- MultiUAV simulation results using HLA communication were able to match prior results using an internal communication scheme. Thus, HLA produced no adverse affects with other MultiUAV simulation components. This work is best summarized in a publication accepted to the 2004 SISO SIW [3].
- A Central Control Module (CCM) was executed in conjunction with MultiUAV that exchanged all HLA vehicle messages. The CCM was based in a DOS console and served as a simple pass-through of all vehicle communication. It demonstrated that through HLA time-management services, the vehicle messages could be filtered through an additional piece of software without affecting the cooperative control algorithms. The long-term objective of the CCM was to manipulate vehicle communication, but this first step provided a seamless blend with the MultiUAV simulation.

At the conclusion of the briefing, AFRL/VA provided feedback and future work directions to ISR:

- AFRL/VA requested the CCM be moved from a DOS console and into a Simulink-based model. This allowed exposure of the vehicle messages in an architecture similar to MultiUAV. This work was accomplished shortly after the briefing and summarized in a research paper accepted at the 2004 AIAA Modeling and Simulation Technology Conference [4].
- AFRL/VA placed additional emphasis on distributing the MultiUAV simulation across multiple machines. Distribution using HLA enabled computational improvements while executing MultiUAV.
- To aid in efficiently executing a distributed version of MultiUAV, AFRL/VA tasked ISR with building an accompanying utility. This utility allowed a user to remotely start, stop, and execute a distributed MultiUAV model.
- AFRL/VA requested that the HLA interface be flexible enough to handle interfaces to other simulations without the need to recompile the software.

In May 2004, ISR briefed [5] AFRL/VA on progress towards the work directions given from the October 2003 demonstration. In this briefing, the new Simulink-based CCM was introduced and demonstrated. The larger portion of the briefing was dedicated to the distributed MultiUAV capability. MultiUAV was executed in a split fashion by placing the target simulation and vehicle simulation on two separate machines. Simulation results were identical to prior results on a single computer and an increase in computational speed was shown. To synchronously start and execute both the target and vehicle simulations, a Java-based Master Control executable was created and used remotely during the demonstration. The initial attempt at a reconfigurable HLA interface S-Function was also shown. The new S-Function dynamically changed its Simulink block interface based upon the HLA message architecture dictated in the Object Model Development (OMD) file.

MultiUAV/JIMM Simulation

In the spring of 2004, AFRL/VA decided the MultiUAV-HLA capability was mature enough to begin interfacing it with other HLA compliant simulations within AFRL/VA. AFRL/VA desired to test the MultiUAV cooperative control algorithms in an environment with higher fidelity threat models instead of the simple targets provided internally, thus exposing assumptions made as the algorithms were developed. An Integrated Air Defense System (IADS) housed in the Joint Integrated Mission Model (JIMM) was chosen to replace the targets in MultiUAV. Since JIMM had been included in the Air Force Standard Toolkit (AFSAT) and HLA connections were viable through AFRL/VA in-house tools, using it in conjunction with MultiUAV was a logical fit.

The initial simulation requirement was to engage the IADS in JIMM with the MultiUAV LOCAAS munitions. Since the Surface-to-Air (SAM) missile sites in the IADS could fire at the UAVs, the munitions were given the unrealistic capability to fire a weapon from a stand-off distance. This allowed the cooperative control algorithms the ability to fully plan and carry-out the mission. During integration, the MultiUAV-HLA interface continued to gain flexibility through software revisions and expandability of the OMD file parser. A prototype MultiUAV/JIMM simulation was briefed and demonstrated at AFRL/VA facilities in September 2004. A research publication accepted at the 2005 SISO SIW provides further details on this initial prototype [6].

To finish the MultiUAV/JIMM simulation work, AFRL/VA requested that a new vehicle and sensor model be implemented in the MultiUAV tool. To accurately engage an IADS in JIMM, a vehicle closer to an Unmanned Combat Aerial Vehicle (UCAV) was required. Leveraging work done from the AAR project, the Joint Unmanned Combat Air Systems Equivalent (JUCAS-EQ) model was implemented as an additional vehicle in MultiUAV. In additional to a new vehicle, MultiUAV also needed a Battle Damage Assessment (BDA) sensor appropriate for verifying the destruction of various SAM sites. To facilitate this requirement, a Synthetic Aperture Radar (SAR) sensor was added as an additional sensor resource in MultiUAV. This work is best summarized in a publication accepted to the 2005 AIAA Guidance, Navigation and Control conference [7].

Once the UCAV and SAR sensor was implemented into MultiUAV, an accurate SEAD mission could be simulated. This capability allowed AFRL/VA to simulate an extra scenario with MultiUAV in addition to the WASD mission. Along with this new mission capability, the HLA interface matured enough to be fully configurable without any need to recompile underlying C++ source code. The final software was delivered and installed by ISR at AFRL/VA facilities in December 2005. AFRL/VA was also supplied with an installation, software design, and user guide of the HLA interface [8].

Upon invitation, ISR presented this new capability at a MultiUAV session hosted at the 2005 American Control Conference [9] and at the 2005 JIMM User's Group Meeting [10].

MultiUAV/FLAMES Simulation

In the spring of 2005, various AFRL/VA simulation efforts were transitioning away from JIMM and to the Flexible Analysis and Modeling Exercise System (FLAMES) platform. FLAMES provides a simpler interface to design and execute various military simulations compared to software such as JIMM. Since AFRL/VA possessed an IADS simulation in FLAMES and modifying the JIMM scenarios proved to be very difficult, a FLAMES remote interface to MultiUAV was requested.

FLAMES provides an Interactive Client and Interactive Server Option if a simulation such as MultiUAV wants not to migrate the existing model entirely into FLAMES. This option includes a set of services that allow external systems to interact with the FLAMES kernel over a network connection. For this effort, the HLA connection in MultiUAV was disabled and a MultiUAV Remote Client Interface (RCI) was built to utilize the services provided in the FLAMES Interactive Client Option. This allowed each vehicle residing in MultiUAV to act as a physical entity in the FLAMES IADS scenario. Though FLAMES provides an HLA connection, it proved too costly and cumbersome to use compared to the Interactive Server/Client system. Additional software was written for the FLAMES simulation to handle the UCAV vehicle properties, precision guided weapons, and SAR sensor. This work is best summarized in a publication accepted to the 2006 Winter Simulation Conference [11].

In November 2005, ISR briefed [12] and installed the MultiUAV/FLAMES simulation at AFRL/VA facilities. AFRL/VA was also supplied an installation, software design, and user guide of the MultiUAV RCI to FLAMES [13].

MultiUAV/FLAMES Electronic Warfare (EW) Simulation

Once the initial MultiUAV/FLAMES integration effort was complete, AFRL/VA noted that added complexity was needed in the SEAD mission tasks solved by MultiUAV. An easy way to challenge the cooperative control algorithms was to reduce the weapon range of the UCAV and increase the weapon range of the SAM. In order to safely release a weapon at a SAM site, the attacking vehicle required an additional vehicle to reduce the weapon effectiveness of the SAM site through electronic jamming.

This task required significant improvements to the trajectory planning software in MultiUAV along with modifications to the jamming models provided in FLAMES. In order to have vehicles perform sequential tasks, loitering routes were built into the trajectory planner to deliberately stall an aircraft if the cooperative control algorithms demanded a later task arrival time. In addition to the loiter route, a circular jam route was also added to allow the UAV to continuously apply a jamming signal to the respective SAM site. The MultiUAV RCI was also modified to interact with the vehicle's jammer equipment model in FLAMES.

In addition to the jamming task, the trajectory planner was also modified to account for no-fly zones. These zones are defined by the maximum distance a SAM can fire a

weapon. If the trajectory planner sensed that a vehicle may intersect a no-fly zone during task assignment, the path was routed accordingly to avoid the zone. These trajectory modifications along with other EW mission considerations were presented [14] and delivered to AFRL/VA in July 2006.

2.2.2 Algorithm Research

In addition to performing MultiUAV software modifications, ISR also investigated various coordinated control algorithms to enable a team of UAVs to accomplish together what none of the alone could accomplish. The objective was to research, develop, and implement algorithms for the cooperative control of multiple UAVs as they collaborate autonomously to complete an assigned mission in both Suppression of Enemy Air Defense (SEAD) and EW scenarios. The team studied, implemented, and compared several different types of algorithms during the May 2005 to December 2006 timeframe. The algorithms were all coded in MATLAB and were integrated into the MultiUAV simulation software for testing. Testing of these algorithms occurred in both the MultiUAV-JIMM environment as well as the MultiUAV-FLAMES environment.

The first type of algorithm examined, implemented and tested was the Mixed Integer Linear Programming (MILP) algorithm. Significant work had already been completed on this topic by Schumacher, Chandler, Pachter, and Pachter [15, 16, 17]. Schumacher et al used MILP to solve the optimal task assignment and timing problem of a team of homogeneous non-survivable UAVs detecting, classifying, attacking, and verifying the destruction of enemy ground targets in a Wide Area Search and Destroy (WASD) mission. Our work in this area [18] extended previous work using MILP to efficiently assign vehicles in a SEAD mission by adding important new assumptions that increased the complexity of the task allocation problem. The new assumptions included: survivable vehicles modeled after JUCAS, heterogeneous teams, and dynamic target discovery. With the addition of a new vehicle and mission type, constraints formulated in the previous MILP research were modified or replaced. A dynamic constraint builder was constructed in MATLAB and integrated into the MultiUAV research tool. This allowed the new tasking and timing constraints to be formulated based upon the number of mission specified vehicles and the number of targets that have either been detected or partially prosecuted. These constraints are easily altered based upon rules designated throughout the MATLAB software. Integration into MultiUAV allowed this approach to be tested in a dynamic environment. The simulation results showed a successful implementation of this MILP task allocation solution for a team of UAVs.

The next algorithm examined, implemented and tested was the Genetic Algorithm (GA). The GAs were used to address the same mission scenario with the same mission assumptions as the MILP. ISR's work in the area built on the work of Shima and Schumacher [19]. Shima et al. applied GAs to the task assignment problem and developed an encoding scheme for a feasible solution as a chromosome. Their work in this area first establishes a general scheme for encoding and applying the genetic algorithm, and then provides results that show the GA outperforming the MILP in a subset of mission scenarios. They later expand their GA work to the prosecution of

8

moving targets using teams of heterogeneous UAVs. ISR's work extended previous work using GAs to assign groups of UAVs to ground targets, as well as previous work using MILP for task allocation in a SEAD mission. ISR's research applied GAs to attempt to find an optimal task assignment for teams of UAVs performing a SEAD mission with heterogeneous vehicles prosecuting pop-up threats. As targets are discovered throughout the mission the problem formulation was dynamically updated to meet the new situation and the algorithm was applied to find a new feasible and efficient solution. We compared the results of previous work done on the same mission scenario using MILP to solve the assignment problem. While the MILP gives the optimal solution it requires much more time and computer resources to formulate and solve. Both approaches work very well for small scenarios with few vehicles and few targets. However, when the mission involves more than three vehicles and two targets, the GA provided a feasible solution in a greatly reduced time. Our work studied the GA approach, formulation, implementation, and integration into a high fidelity simulation [20].

The next area of research investigated was the parallelization of the GA utilized for task assignment of a team of UAVs conducting a SEAD mission [21]. The GA was developed and implemented in the MultiUAV simulation environment for testing during the initial phase discussed above. The original (non-parallel version) of the GA demonstrated improved performance over the MILP algorithm. In order to further improve on the GA performance, the algorithm was parallelized with each UAV acting as an independent processor. Two different implementations were explored. The first implementation employed identical algorithms on each processor seeded with a different random number, requiring an exchange of "best solution" at the end. The second utilizes a GA Island Model approach that allowed "good" solutions to migrate between processors during the computation period, necessitating information exchange several times during the evolution of generations. The results of these implementations were compared to the original GA performance.

The Parallelized and Island Model versions of the GA outperform both the original GA formulation and the MILP. However, it is important to mention certain drawbacks. The Parallelized GA comes to a better solution faster, but requires an extra communication step at the end. This can be a problem if intra-team communication must be limited or if communication failures occur. On average, the Island Model shows improved performance over the Parallelized GA, but requires increased communication, thus leaving it more vulnerable to communication errors. The performance increase of the Island Model may not outweigh the demand in communication requirements compared to the Parallelized approach.

The next algorithm investigated was the Simulated Annealing (SA) algorithm [22]. ISR researched, developed, and implemented the SA algorithm applied to the task allocation of multiple UAVs for an EW mission. The EW mission has the following basic assumptions: a priori knowledge of the battlespace, the vehicles are all UCAVs but are outfitted with different payloads, vehicles have sensors that can allow them to accomplish the classify task out of harms way before jamming starts, vehicles outfitted with jamming

9

equipment are only tasked to do jamming, and vehicles with weapons payloads also have sensors for classifying and doing battle damage assessment on targets. Both the GA and the new SA were applied to the EW mission scenario. After encoding the problem and implementing the algorithms in MATLAB, two different types of tests were performed to compare the two algorithms. The performance analysis was based on the quality of solutions in a fixed period of time and time to convergence. For our implementations, all tests indicate that the genetic algorithm is a superior algorithm to the simulated annealing algorithm for this problem domain. The algorithms were implemented using standard practice for each, however tuning of the algorithms may produce somewhat different results.

2.3 Infinity Cube Flight Simulator

The Infinity Cube Flight Simulators are state of the art PC-based flight simulators that were installed at Wright-Patterson AFB and ISR's Fairmont, WV facility with the intention of investigating the feasibility of conducting low cost joint remote simulations. To conduct joint remote simulations, a method of communicating simulation data between the two simulators was established. For this purpose an ISDN line was leased. The protocols for transmitting the data that were investigated are Transmission Control Protocol (TCP) and User Datagram Protocol (UDP).

Preliminary analysis showed that TCP would be the protocol of choice as it mandates that packets arrive in a prescribed order; however, the delay penalty associated with re-transmitting lost or damaged packets quickly outweighed the positive factors. UDP communication does not guarantee the order the packets will be received, or for that matter, that a packet will ever be received. However, by sending a packet only once, large delays inherent with re-transmission can be avoided..

Since maximizing execution speed was the primary goal, a decision was made to process the first packet received, provided it originated later than the previous packet processed. This method would have worked well if packets were only missing occasionally and sequential packets weren't lost or didn't arrive out of sequential order. Experimentation showed that this assumption was false. The simulation was very jumpy. To fix the gaps in information that were created by out-of-order or missing packets, dead reckoning software was implemented to bridge the gaps in navigational information. This smoothed out the performance and enabled the joint remote simulation to operate adequately.

Given that the aircraft position data is not always exact due to the dead reckoning procedure implemented, there is limited accuracy that can be achieved based on the amount of packet loss. Even with this error, this simulation method would have been feasible for many applications requiring a remote simulation. One limit to the applicability of this simulation environment is formation flight. This simulation was envisioned for use on the AAR project. It would enable a tanker, UAV, and boom operator to perform simulated refueling from remote locations. With this communication method, a packet's one way trip was clocked at approximately 60 milliseconds, which at

240 knots corresponded to an unacceptable simulated distance error. This magnitude of error would be prohibitive for an accurate refueling, or formation flying simulation.

3 CONCLUSIONS

The range of research and system development achieved during the contract period of performance required a broad range of skill sets and techniques. The early focus of the program was the development of the MultiUAV simulation tool. Modifications to the tool remained a primary focus through the first few years of the program. Once the tool stabilized, the team shifted their multiple UAV coordinated control focus from tool development to algorithm development. Many of the algorithms investigated built upon early AFRL/VACA work and were evaluated in new missions. Feasibility and promise was shown for most of the algorithms in some the more complex missions.

The AAR research was primarily accomplished by ISR's subcontractor, WVU, in the early years of the program. However, once the AAR simulation environment was built, ISR conducted the majority of the MV research and executed the trade studies, the primary focus of the AAR effort. General feature extraction algorithms were evaluated using visible spectrum flight test video at contact position. The results of those evaluations drove the decision to try a long-wave infrared spectrum camera, proved that a general feature extraction algorithm wasn't sufficient to achieve AAR goals, and began ISR on the task of designing specific feature extraction algorithms from scratch. While that work continues on a separate contract, it was not discussed in this report.

Finally, the Infinity Cube Flight Simulator, installed at ISR in April 2004 was transferred to AFRL's Wright-Patterson AFB facilities in September 2006. The Infinity Cube was used for some minor remote simulation integration testing and AAR Learjet HDD development, but was never the primary focus of the ISR research efforts as directed by AFRL.

4 REFERENCES

[1] Griffith, B., Rogers, S., Stolarik, B., and Niland, W., "HLA/MultiUAV Integration Preliminary Design Review," Presented to AFRL/VA, Morgantown, WV, June 2003.

[2] Stolarik, B., Rogers, S., and Niland, W. "HLA/MultiUAV Integration Technical Status Update," Presented to AFRL/VA, Wright Patterson Air Force Base, OH, October 2003.

[3] Stolarik, B., Niland, W., and Givens, B., "Multiple UAV Communication Modeling Using the HLA," *Proceedings of the 2004 SISO Spring Simulation Interoperability Workshop*, Alexandria, VA, 04S-SIW-104, April 2004.

[4] Stolarik, B., Niland, W., and Rasmussen, S., "Enabling Inter-Vehicle Communication Modeling in MultiUAV Using the HLA", *Proceedings of the 2004 AIAA Modeling and Simulation Technologies Conference*, Providence, RI, AIAA 2004-5043, August 2004.

[5] Stolarik, B., and Niland, W., "Research Update 5-14-04," Presented to AFRL/VA, Wright Patterson Air Force Base, OH, May 2004.

[6] Niland, W., Stolarik, B., Rasmussen, S., Allen, K., and Finley, K., "Enhancing a Collaborative UAV Mission Simulation Using JIMM and the HLA", *Proceedings of the 2005 SISO Spring Simulation Interoperability Workshop*, San Diego, CA, 05S-SIW-078, April 2005.

[7] Niland, W., Stolarik, B., Harman, J., Petersavage, J., and Rasmussen, S., "The Inclusion of a Supplementary Mission Scenario into the MultiUAV Research Tool," *Proceedings of the 2005 AIAA Guidance, Navigation and Control Conference and Exhibit*, San Francisco, CA, AIAA 2005-6479, August 2005.

[8] Institute for Scientific Research, Inc., "The High Level Architecture MultiUAV Interface User's Guide," U.S. Air Force Research Laboratory, Wright Patterson Air Force Base, OH, December 2004.

[9] Niland, W., and Stolarik, B., "The HLA Toolbox for MultiUAV," Presented at the 2005 American Control Conference, Portland, OR, June 2005.

[10] Niland, W., Stolarik, B., Rasmussen, S., Allen, K., and Finley, K., "Enhancing a Collaborative UAV Mission Simulation Using JIMM and the HLA," Presented at the 2005 JIMM User's Group Meeting, Alexandria, VA, May 2005.

[11] Niland, W., "The Migration of a Collaborative UAV Testbed into the FLAMES Simulation Environment", *Proceedings of the 2006 Winter Simulation Conference*, Monterey, CA, December 2006.

[12] Niland, W., and Stolarik, B., "MultiUAV Remote Client Interface for FLAMES," Presented to AFRL/VA, Wright Patterson Air Force Base, OH, November 2005.

[13] Institute for Scientific Research, Inc., "MultiUAV Remote Client Interface for FLAMES User Guide," U.S. Air Force Research Laboratory, Wright Patterson Air Force Base, OH, November 2005.

[14] Darrah, M., Niland, W., Stolarik, B., Petersavage, J., Hill, A., Walp, L., "Task Allocation Approaches for UCAV Teams in Electronic Warfare SEAD Missions," Presented to AFRL/VA, Wright Patterson Air Force Base, OH, July 2006.

[15] Schumacher, C., Chandler, P., Pachter, M. and Pachter, L., "Constrained Optimization for UAV Task Assignment," *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit*, Providence, RI, August 2004.

[16] Schumacher, C., Pachter, M., Chandler, P., and Pachter, L., "UAV Task Assignment with Timing Constraints," *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit*, Austin, TX, August 2003.

[17] Schumacher, C., Chandler, P., and Rasmussen, S., "Task Allocation for Wide Area Search Munitions with Variable Path Lengths*," Proceedings of the American Control Conference*, Denver, CO, June 2003.

[18] Darrah M., Niland W., and Stolarik B., "Multiple UAV Dynamic Task Allocation using Mixed Integer Linear Programming in a SEAD Mission," *Proceedings of the AIAA Infotech@Aerospace Conference*, Alexandria, VA, AIAA 2005-7164, September 2005.

[19] Shima, T. S. and Schumacher, C., "Assignment of Cooperating UAVs to Simultaneous Tasks using Genetic Algorithms," AIAA Guidance, Navigation, and Control Conference and Exhibit, AIAA 2005-5829, San Francisco, California, 2005.

[20] Darrah M., Niland W., Stolarik B., and Walp L., "UAV Cooperative Task Assignments for a SEAD Mission using Genetic Algorithms," *Proceeding of Guidance, Navigation and Control Conference*, Keystone, CO, August 2006.

[21] Darrah M. and Niland W., "Increasing UAV Task Assignment Performance through Parallelized Genetic Algorithms," *Proceedings of Infotech@Aerospace Conference,* Rohnert Park, California, May 2007.

[22] Darrah M., Niland W. and Stolarik B., "Multiple UAV Task Allocation for an Electronic Warfare Mission Comparing Genetic Algorithms and Simulated Annealing," To be submitted.